

AMENDMENTSIN THE SPECIFICATION:

On page 6 of the specification, please delete paragraphs **0019** and **0020**.

On pages 7 and 8 of the specification, please amend paragraph **0024** as follows:

**[0024]** To implement an embodiment of the method, referring to FIG. 1, a supervisory user inputs a Template 2 for each type of communication message desired, along with individual communication 3 components (data files) that can be selected and inserted into the Template. The supervisory user creates programs for enabling the Special Processing Instructions 4, i.e., instructions for storing the communication components. Typically, these programs, Template and data files are stored in a repository 5, and are accessed and protected by a password or security protocol. The repository is typically a computer memory device such as a hard drive, operably connected to a programmed central processor, and thereby to random access memory, input devices such as scanners, keyboards, or mouse, output devices such as a display monitor or printer, and, optionally, via a cable, modem or network link to separate computer workstations. The term repository as used herein is intended to include a central repository, a distributed repository and a central and distributed repository. The program receives instructions from users 6 and implements those instructions to create business communications output by retrieving the appropriate component(s), Template (with Special Processing Instructions) from the memory device, and applying the components to the Template. The Template's Special Processing Instructions direct the component to the appropriate location and/or sequence in the communication, and the output is generated and saved as a separate file or remain in transient

form, as is the case with a dynamic web page, sound communication, or video file format. Again, these business communications can include, but are not limited to e-mails, faxes, letters or web pages.

On page 16 of the specification, please amend paragraph **0043** as follows:

**[0043]** The above mentioned Special Processing Instruction files may be created by programmers, supervisors or users with appropriate security access in program files that are a part of, or separate from but accessibly located to, the program of the invention. These instructions may be defined by the company's supervisory users, and programmed in a machine or computer language which is readable by the database and its ancillary programs. The following examples illustrate the instructions that may be contained in Special Processing Instruction files which build and route the communications and reset the session variables:

FILE 1

```
<%
'turn off page caching in users browser
Response.Expires=0
Response.Buffer=True

'define session wide variables

Session ("SecurityDatabase")="Provider=Microsoft.Jet.OLEDB.3.51;Persist Security
Info=False;Data Source=e:\indever\data\private\QSecurity\QSecurity.mdb"
Session("SecurityLog")="Provider=Microsoft.Jet.OLEDB.3.51;Persist Security
Info=False;Data Source=e:\indever\data\private\QSecurity\QSecurity.log"

Session("QServicesDatabase")="Provider=Microsoft.Jet.OLEDB.3.51;Persist Security
Info=False;Data Source=e:\indever\data\private\QServices\QServices.mdb"
Session("QServicesLog")="Provider=Microsoft.Jet.OLEDB.3.51;Persist Security
Info=False;Data Source=e:\indever\data\private\QServices\QServices.log"

Session("QForumDatabase")="Provider=Microsoft.Jet.OLEDB.3.51;Persist Security
Info=False;Data Source=e:\indever\data\private\QForum\QForum.mdb"
Session("QForumLog")="Provider=Microsoft.Jet.OLEDB.3.51;Persist Security
```

```
Info=False;Data Source=e:\indever\data\private\QForum\QForum.log"

Session("QClientDatabase")="Provider=Microsoft.Jet.OLEDB.3.51;Persist Security
Info=False;Data Source=e:\indever\data\private\QClient\QClient.mdb"
Session("QClientLog")="Provider=Microsoft.Jet.OLEDB.3.51;Persist Security
Info=False;Data Source=e:\indever\data\private\QClient\QClient.log"

Session("QResourcesDatabase")="Provider=Microsoft.Jet.OLEDB.3.51;Persist
Security Info=False;Data
Source=e:\indever\data\private\QResources\QResources.mdb"
Session("QResourcesLog")="Provider=Microsoft.Jet.OLEDB.3.51;Persist Security
Info=False;Data Source=e:\indever\data\private\QResources\QResources.log"

Session("QCompetition")="Provider=Microsoft.Jet.OLEDB.3.51;Persist Security
Info=False;Data Source=e:\indever\data\private\QCompetition\QCompetition.mdb"
Session("QCompetitionLog")="Provider=Microsoft.Jet.OLEDB.3.51;Persist Security
Info=False;Data Source=e:\indever\data\private\QCompetition\QCompetition.log"

Session("QPlanDatabase")="Provider=Microsoft.Jet.OLEDB.3.51;Persist Security
Info=False;Data Source=e:\indever\data\private\QPlan\QPlan.mdb"
Session("QPlanLog")="Provider=Microsoft.Jet.OLEDB.3.51;Persist Security
Info=False;Data Source=e:\indever\data\private\QPlan\QPlan.log"

Session("QManageDatabase")="Provider=Microsoft.Jet.OLEDB.3.51;Persist Security
Info=False;Data Source=e:\indever\data\private\QManage\QManage.mdb"
Session("QManageLog")="Provider=Microsoft.Jet.OLEDB.3.51;Persist Security
Info=False;Data Source=e:\indever\data\private\QManage\QManage.log"

set qlog = server.createobject("qlib.qlog")

Dim qweb, qreport
dim soutput
dim sResult
dim sFileName

select case Request.Form("QWebAction")

case "Login"
'Display Login

Session("QWebMemberID")=Request.Form("ID")
Session("QWebMemberPassword")=Request.Form("Password")
Session("SecurityID")=Request.Form("Identity")

'qlog.entry "(" & Session("QWebMemberID") & ") - Logging In . . .",
Session("SecurityLog")
```

```
set QReport = Server.Createobject("Qlib.QReport")
sResult = QReport.PublishPage(
Session("SecurityDatabase"), _
"SELECT * FROM Security WHERE ID = ' " & Request.Form("ID") & " ' AND
Password = ' " & Request.Form("Password") & " ' AND SecurityID=" &
Request.Form("Identity"), _
Session("SecurityDatabase"), _
"SELECT * FROM Templates WHERE TemplateID = 'Desktop' ",_
"Template", _
" ")
IF sResult = " " THEN
sResult = QReport.PublishPage(
Session("SecurityDatabase"), _
"SELECT * FROM Dummy",_
Session("SecurityDatabase"), _
"SELECT * FROM Templates WHERE TemplateID = 'AccessDenied' ",_
"Template", _
" ")
Response.Write(sResult)
ELSE
Response.Write(sResult)
END IF

set QReport = nothing
end select

set qlog = nothing
%>
```

FILE 2

```
<%
**** has user session been validated?
'if Session ("QWebMemberID") = NULL then
    'Response.Redirect "login.htm"
'end if

'turn off page caching in users browser
Response.Expires=0
Response.Buffer=True

set qlog = server.createobject (" qlib.qlog")
```

```
Dim qweb, qreport
dim sOutput
dim sResult
dim sFileName
dim dothing

' *** Execute an Recordsetless SQL Command on the Database ***
Function ExecuteSQL2 (SQL)
    dim cn

    '*** open a database connection
    set cn = Server.CreateObject(" ADODB.Connection")
    cn.open Session("SecurityDatabase")

    '*** execute sql command
    cn.Execute SQL

    '*** close connection
    cn.Close
    set cn = nothing

End Function

Set dothing=Request.QueryString (" QWebAction")
select case dothing

case "ViewPage"
'View Page
    set QReport = Server.Createobject("Qlib.QReport")

        sResult = QReport.PublishPage( _
        Session("SecurityDatabase"), _
        "SELECT * FROM " & Request.QueryString("QWT") & " WHERE " &
        Request.QueryString("QWF") & " = " & Request.QueryString("QWFV") & " ORDER
        BY "
        & Request.QueryString("QWS") ,_
        Session ("SecurityDatabase") ,_
        "SELECT * FROM Templates WHERE TemplateID =' " &
        Request.QueryString("QwebID") & " ' ,_
        "Template", _
        " ")

        Response.Write(sResult)

    set QReport = nothing
```

```
case "Greeting"
'Greeting

    set QReport = Server.Createobject("Qlib.Qreport")

        sResult = QReport.PublishPage( _
        Session("SecurityDatabase") , __
            "SELECT * FROM " & Request.QueryString("QWT") & " WHERE
SecurityID=" & Session("SecurityID") & " AND " & Request.QueryString("QWF") & "="
" &
Request.QueryString("QWFV") & " ORDER BY " & Request.QueryString("QWS") ,_
            Session("SecurityDatabase") , __
            "SELECT * FROM Templates WHERE TemplateID = ' " &
Request.QueryString("QWebID") & " ' , __
            "Template" , __
            " ")

        Response.Write(sResult)
        'Update last log on
        ExecuteSQL2 "UPDATE Security SET LastLogOn = #" & Now()
        & "# WHERE SecurityID=" & Session("SecurityID")
        set QReport = nothing

case "PersonalDesktop"
'PersonalDesktop

    set QReport = Server.Createobject("Qlib.QReport")

        sResult = QReport.PublishPage( _
        Session("SecurityDatabase") , __
            "SELECT      *      FROM      UserDesktops      WHERE      SecurityID="      &
Session("SecurityID") &
"AND Status='Active' ORDER BY SORTFIELD" , __
            Session("SecurityDatabase") , __
            "SELECT * FROM Templates WHERE TemplateID = 'PERSONALDESKTOP' "
, __
            "Template" , __
            " ")

        Response.Write(sResult)

    set QReport = nothing

end select

    set qlog = nothing

%>
```

Separate, marked up pages showing the changes made are enclosed.

IN THE DRAWINGS:

Please delete FIGs. 8 and 9.

**REMARKS**

The amendments are made for clarification of the correlations between the drawings and text, and in response to the statement in the Notice to File Corrected Application Papers that the drawings contain excessive text. The text contained in former drawings 8 and 9 has been moved to the specification. No new matter is added.

If the Examiner believes, for any reason, that personal communication will expedite prosecution of this application, I ask that the Examiner telephone me directly.

**AUTHORIZATION**

The Commissioner is hereby authorized to charge any additional fees, including any fees for net addition of claims, which may be required for this Amendment, or credit any overpayment to deposit account No. 50-0436. Applicants believe that no extension of time is necessary, however, in the event that an extension of time may be required, the Commissioner is requested to grant a petition for that extension of time which is required to make this response timely and is hereby authorized to charge any fee for such an extension of time or credit any overpayment for an extension of time to deposit account No. 50-0436.

Respectfully submitted,

PEPPER HAMILTON LLP



James M. Singer  
Registration No. 45,111

Pepper Hamilton LLP  
50<sup>th</sup> Floor, One Mellon Center  
500 Grant Street  
Pittsburgh, PA 15219-2505  
Phone: 412-454-5000  
Facsimile: 412-281-0717  
**Date: May 22, 2002**

**MARKED VERSION SHOWING CHANGES MADE****IN THE SPECIFICATION:**

On page 6 of the specification, paragraphs **0019** and **0020** are deleted.

On pages 7 and 8 of the specification, paragraph **0024** is amended as follows:

**[0024]** To implement an embodiment of the method, referring to FIG. 1, a supervisory user inputs a Template 2 for each type of communication message desired, along with individual communication 3 components (data files) that can be selected and inserted into the Template. The supervisory user creates programs for enabling the Special Processing Instructions 4, i.e., instructions for storing the communication components. Typically, these programs, Template and data files are stored in a repository 5, and are accessed and protected by a password or security protocol. The repository is typically a computer memory device such as a hard drive, operably connected to a programmed central processor, and thereby to random access memory, input devices such as scanners, keyboards, or mouse, output devices such as a display monitor or printer, and, optionally, via a cable, modem or network link to separate computer workstations. The term repository as used herein is intended to include a central repository, a distributed repository and a central and distributed repository. The program receives instructions from users 6 and implements those instructions to create business communications output by retrieving the appropriate component(s), Template (with Special Processing Instructions) from the memory device, and applying the components to the Template. The Template's Special Processing Instructions direct the component to the appropriate location and/or sequence in the

communication, and the output is generated and saved as a separate file or remain in transient form, as is the case with a dynamic web page, sound communication, or video file format. Again, these business communications can include, but are not limited to e-mails, faxes, letters or web pages.

On page 16 of the specification, paragraph **0043** is amended as follows:

**[0043]** The above mentioned Special Processing Instruction files may be created by programmers, supervisors or users with appropriate security access in program files that are a part of, or separate from but accessibly located to, the program of the invention. These instructions may be defined by the company's supervisory users, and programmed in a machine or computer language which is readable by the database and its ancillary programs. [FIG. 8 and FIG. 9 depict examples of] The following examples illustrate the instructions that may be contained in Special Processing Instruction files which build and route the communications and reset the session variables: [.]

#### FILE 1

```

<%
'turn off page caching in users browser
Response.Expires=0
Response.Buffer=True

'define session wide variables

Session ("SecurityDatabase")="Provider=Microsoft.Jet.OLEDB.3.51;Persist Security
Info=False;Data Source=e:\indever\data\private\QSecurity\QSecurity.mdb"
Session("SecurityLog")="Provider=Microsoft.Jet.OLEDB.3.51;Persist Security
Info=False;Data Source=e:\indever\data\private\QSecurity\QSecurity.log"

Session("QServicesDatabase")="Provider=Microsoft.Jet.OLEDB.3.51;Persist Security
Info=False;Data Source=e:\indever\data\private\QServices\QServices.mdb"
Session("QServicesLog")="Provider=Microsoft.Jet.OLEDB.3.51;Persist Security
Info=False;Data Source=e:\indever\data\private\QServices\QServices.log"

```

Session("QForumDatabase")="Provider=Microsoft.Jet.OLEDB.3.51;Persist Security Info=False;Data Source=e:\indever\data\private\QForum\QForum.mdb"  
Session("QForumLog")="Provider=Microsoft.Jet.OLEDB.3.51;Persist Security Info=False;Data Source=e:\indever\data\private\QForum\QForum.log"

Session("QClientDatabase")="Provider=Microsoft.Jet.OLEDB.3.51;Persist Security Info=False;Data Source=e:\indever\data\private\QClient\QClient.mdb"  
Session("QClientLog")="Provider=Microsoft.Jet.OLEDB.3.51;Persist Security Info=False;Data Source=e:\indever\data\private\QClient\QClient.log"

Session("QResourcesDatabase")="Provider=Microsoft.Jet.OLEDB.3.51;Persist Security Info=False;Data Source=e:\indever\data\private\QResources\QResources.mdb"  
Session("QResourcesLog")="Provider=Microsoft.Jet.OLEDB.3.51;Persist Security Info=False;Data Source=e:\indever\data\private\QResources\QResources.log"

Session("QCompetition")="Provider=Microsoft.Jet.OLEDB.3.51;Persist Security Info=False;Data Source=e:\indever\data\private\QCompetition\QCompetition.mdb"  
Session("QCompetitionLog")="Provider=Microsoft.Jet.OLEDB.3.51;Persist Security Info=False;Data Source=e:\indever\data\private\QCompetition\QCompetition.log"

Session("QPlanDatabase")="Provider=Microsoft.Jet.OLEDB.3.51;Persist Security Info=False;Data Source=e:\indever\data\private\QPlan\QPlan.mdb"  
Session("QPlanLog")="Provider=Microsoft.Jet.OLEDB.3.51;Persist Security Info=False;Data Source=e:\indever\data\private\QPlan\QPlan.log"

Session("QManageDatabase")="Provider=Microsoft.Jet.OLEDB.3.51;Persist Security Info=False;Data Source=e:\indever\data\private\QManage\QManage.mdb"  
Session("QManageLog")="Provider=Microsoft.Jet.OLEDB.3.51;Persist Security Info=False;Data Source=e:\indever\data\private\QManage\QManage.log"

set qlog = server.createobject("qlib.qlog")

Dim qweb, qreport  
dim soutput  
dim sResult  
dim sFileName

select case Request.Form("QWebAction")

case "Login"  
'Display Login

Session("QWebMemberID")=Request.Form("ID")  
Session("QWebMemberPassword")=Request.Form("Password")

```

Session("SecurityID")=Request.Form("Identity")

'qlog.entry "(" & Session("QWebMemberID") & ")" - Logging In . . ,
Session("SecurityLog")

set QReport = Server.Createobject("Qlib.QReport")
sResult = QReport.PUBLISHPAGE(
Session("SecurityDatabase"),
"SELECT * FROM Security WHERE ID = ' " & Request.Form("ID") & " ' AND
Password = ' " & Request.Form("Password") & " ' AND SecurityID=" &
Request.Form("Identity"),
Session("SecurityDatabase"),
"SELECT * FROM Templates WHERE TemplateID = 'Desktop' ",
"Template",
" ")
IF sResult = " " THEN
sResult = QReport.PUBLISHPAGE(
Session("SecurityDatabase"),
"SELECT * FROM Dummy",
Session("SecurityDatabase"),
"SELECT * FROM Templates WHERE TemplateID = 'AccessDenied' ",
"Template",
" ")
Response.Write(sResult)
ELSE
Response.Write(sResult)
END IF

set QReport = nothing

end select

set qlog = nothing

%>

FILE 2

<%
'*** has user session been validated?
'if Session ("QWebMemberID") = NULL then
    'Response.Redirect "login.htm"
'end if

'turn off page caching in users browser

```

```
Response.Expires=0
Response.Buffer=True

set qlog = server.createobject (" qlib.qlog")

Dim qweb, qreport
dim sOutput
dim sResult
dim sFileName
dim dothing

' *** Execute an Recordsetless SQL Command on the Database ***
Function ExecuteSQL2 (SQL)
    dim cn

    *** open a database connection
    set cn = Server.CreateObject(" ADODB.Connection")
    cn.open Session("SecurityDatabase")

    *** execute sql command
    cn.Execute SQL

    *** close connection
    cn.Close
    set cn = nothing

End Function

Set dothing=Request.QueryString (" QWebAction")
select case dothing

case "ViewPage"
'View Page
    set QReport = Server.Createobject("Qlib.QReport")

    sResult = QReport.PublishPage(
        Session("SecurityDatabase"),
        "SELECT * FROM " & Request.QueryString("QWT") & " WHERE " &
        Request.QueryString("QWF") & " = " & Request.QueryString("QWFV") & " ORDER
        BY "
        & Request.QueryString("QWS"),
        Session ("SecurityDatabase"),
        "SELECT * FROM Templates WHERE TemplateID =' " &
        Request.QueryString("QwebID") & " '",
        "Template",
        " ")

```

```
Response.Write(sResult)
set QReport = nothing
case "Greeting"
'Greeting
set QReport = Server.Createobject("Qlib.Qreport")
sResult = Qreport.PublishPage(
Session("SecurityDatabase"),
"SELECT * FROM " & Request.QueryString("QWT") & " WHERE
SecurityID=" & Session("SecurityID") & " AND " & Request.QueryString("QWF") & "="
" &
Request.QueryString("QWFV") & " ORDER BY " & Request.QueryString("QWS"),
Session("SecurityDatabase"),
"SELECT * FROM Templates WHERE TemplateID = ' " &
Request.QueryString("QWebID") & " '",
"Template",
" ")
Response.Write(sResult)
'Update last log on
ExecuteSQL2 "UPDATE Security SET LastLogOn = #" & Now()
& "# WHERE SecurityID=" & Session("SecurityID")
set QReport = nothing
case "PersonalDesktop"
'PersonalDesktop
set QReport = Server.Createobject("Qlib.QReport")
sResult = QReport.PublishPage(
Session("SecurityDatabase"),
"SELECT * FROM UserDesktops WHERE SecurityID=" &
Session("SecurityID") &
"AND Status='Active' ORDER BY SORTFIELD",
Session("SecurityDatabase"),
"SELECT * FROM Templates WHERE TemplateID = 'PERSONALDESKTOP' "
"Template",
" ")
Response.Write(sResult)
set QReport = nothing
end select
```

set qlog = nothing

%>

IN THE DRAWINGS:

FIGs. 8 and 9 are deleted.